

УДК 004.384

## **ФИЛОСОФСКИЙ АСПЕКТ ПРОГРАММИРОВАНИЯ**

Лындов А.И., магистрант гр. ПИМ-181, I курс  
Научный руководитель: Баумгартэн М. И к.ф.-м.н., доцент, доцент  
Кузбасский государственный технический университет  
имени Т.Ф. Горбачева  
г. Кемерово

Программирование зародилось в двадцатом веке как элемент, как часть кибернетики. Программирование так же, как и кибернетика, междисциплинарна, поскольку основы программирования были положены из целого ряда наук, таких как: математика, физика, электротехника, логика, лингвистика. В свою очередь фундаментом для появления кибернетики стал ряд научных и естественных достижений в таких областях как теория автоматического управления и радиоэлектроники, что позволило разработать высокоскоростные вычислительные устройства.

Вначале, когда программы для таких устройств были довольно просты, ими занимались сами производители компьютеров. Но с увеличением сложности и увеличением объема программ люди стали нуждаться в специальной подготовке и, что наиболее важно, в особом мышлении, которое могло бы «вдохнуть жизнь» во все более интеллектуальный электронный «мозг». Затем они стали рассматривать написание программ для электронных компьютеров как особый вид интеллектуальной деятельности, который иногда сравнивают с искусством.

Рассмотрим вопрос составления программ для таких программно-управляемых вычислительных устройств как персональные компьютеры, которые наиболее доступны широкому кругу людей и которые хорошо знакомы с таким термином как “интеллектуальное программирование”. Термин “интеллектуальное программирование” (ИП) здесь будет пониматься как составление таких программ, которые могут соперничать с человеческим разумом или помогать ему в решении интеллектуальных задач [1]. ИП по сути представляет одно из основных направлений в искусственном интеллекте. Из него вытекают такие области исследований как: “автоматический синтез программ”, “системы когнитивной практики”, “инструментальные системы” и т.д.

Философия и информатика тесно связаны в современной философии. Степень актуальности этой связи может варьироваться в зависимости от философской традиции, подхода, области, в которой мы находимся, и проблемы, к решению которой мы стремимся. Однако неоспоримым фактом является то, что связь между философией и информатикой углубилась и развивалась, по крайней мере, после работы Алана Тьюринга [5]. Дискуссии по искусственному интеллекту тесно связаны с философскими дискуссиями о понятии ин-

теллекта; проблемы вычислений напрямую связаны с дискуссиями в логике, философии логики и философии математики; вычислительные модели оказывают огромное влияние на когнитивные науки, философию ума и даже философию науки. Конечно, можно привести и другие примеры.

Существует, по крайней мере, несколько различных способов соединения философии и информатики. С одной стороны, когда понятия, пришедшие из информатики, используются в качестве входных данных для определенных философских размышлений, как это имеет место в философии сознания и философии математики. Колберн [6] дает общее обсуждение связи между философией и информатикой. С другой стороны, когда концепции, относящиеся к этой дисциплине, рассматриваются с мета-теоретической точки зрения с целью выяснения аспектов этих концепций, которые выпадают из дисциплинарного обсуждения, даже тогда эти аспекты имеют отношение к онтологическому и эпистемологическому статусу дисциплины. [7, 8].

Третий способ решения этой проблемы может быть добавлен к историческим исследованиям в области компьютерных наук, в этом случае мы обязательно нуждаемся в элементах, которые принадлежат философии науки. Проведение исследований по проблемам философии информатики любым из описанных выше способов требует междисциплинарных исследовательских групп.

Философия информатики, понимаемая в этом смысле, принимает некоторые понятия, которые принадлежат к дисциплине и лежат в основе онтологических и эпистемологических дискуссий по этой дисциплине.

Во-первых, потому что ответ на некоторые фундаментальные вопросы определит то, как мы понимаем информатику. Во-вторых, потому, что эти ответы связаны друг с другом, учитывая их конформацию в качестве последовательной теоретической позиции.

Жизненный цикл «электронной мысли».

Сам принцип человеческого мышления построен на сложном механизме взаимодействия нейронов в мозгу человека. Существует прямая аналогия между деятельностью человеческого мозга и работой компьютерной программы. В его работе используются электрические импульсы, проходящие через миллионы транзисторов (аналогично импульсам, проходящим через нейроны). Человеческий мозг также чист при рождении, как компьютер без программы. Взрослея человек приобретает знания, а компьютер, получая более сложную программу, «умнеет». Компьютерные программы уже превосходят людей в такой абстрактной игре, как шахматы. Они могут выполнять как рутинную работу, так и интеллектуальную работу, подсказывая человеку, например, при проектировании чего-либо, готовые решения, выполняя роль экспертов. Но можно ли этот процесс назвать мышлением? Ведь мышление - это активный процесс отражения объективного мира в концепциях, суждениях, теориях и т. д., связанных с решением определенных проблем. [1]

Эпистемологический статус информатики.

Проблема определения эпистемологического статуса информатики возникла, как только информатика стала независимой дисциплиной, отличной от математики, между 1960-ми и 1970-ми годами [2]. С 1970-х годов стало ясно, что информатику следует рассматривать частично как математическую дисциплину, частично как научную дисциплину, а частично как инженерную дисциплину, поскольку в ней используются математические, эмпирические и инженерные методы [3]. Тем не менее, состоялась дискуссия о том, должна ли информатика рассматриваться в основном как математическая дисциплина, отрасль инженерии или как научная дисциплина.

Обладатели математической природы информатики предполагают, что программы - это математические объекты, в отношении которых можно проводить чисто дедуктивные рассуждения, предоставляемые формальными методами теоретической информатики. При условии, что спецификации программы будут расширены на формальном языке, и при условии, что код программы будет представлен на одном и том же формальном языке, формальная семантика обеспечивает средство для подтверждения правильности. Соответственно, знания о поведении вычислительных артефактов приобретаются с помощью дедуктивных рассуждений, связанных с математическими доказательствами правильности.

Причина, лежащая в основе такого рационалистического оптимизма в отношении того, что можно знать о вычислительных системах, заключается в том, что они являются артефактами, то есть системами, созданными человеком, и что таким образом можно с уверенностью предсказать их поведение [4].

Первоначальная мотивация для математического анализа вычислений исходила из математической логики. Его происхождение можно найти в вопросе Гильберта: может ли существовать алгоритм, процедура для решения произвольного предложения логики. Чтобы решить этот вопрос, требовалась строгая модель неформальной концепции эффективного или механического метода в логике и математике. При условии, что это, прежде всего, математическое усилие: необходимо разработать математический аналог нестандартного понятия [1].

### Список литературы:

1. Философские аспекты программирования. [Электронный ресурс] – Режим доступа: <http://progcpp.narod.ru/doc/articles/filosof.htm> (дата обращения 23.03.2019)
2. Tedre, Matti, “Computing as a Science: A Survey of Competing Viewpoints”, *Minds and Machines*, 2011, c: 361–387
3. Tedre, Matti & Eeki Sutinen, “Three Traditions of Computing: What Educators Should Know”, *Computer Science Education*, 2008, c: 153–170
4. Knuth, Donald E., “Computer Science and Its Relation to Mathematics”, *The American Mathematical Monthly*, 1974, c: 323–343.

5. A.Turing, "On Computable Numbers, with an Application to the *Entscheidungsproblem*", *Proceedings of the London Mathematical Society*, Series 2, 42, 1936, c: 230–65
6. A.Turing, "On Computable Numbers, with an Application to the *Entscheidungsproblem*", *Proceedings of the London Mathematical Society*, Series 2, 42, 1936, c: 230–65.
7. R.Turner and A. Eden, "The Philosophy of Computer Science", *Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.), 2011. [Электронный ресурс] – Режим доступа: <http://plato.stanford.edu/archives/win2011/entries/computer-science/> (дата обращения 23.03.2019)
8. R.Turner, "The Philosophy of Computer Science", *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.), 2014. [Электронный ресурс] – Режим доступа: <http://plato.stanford.edu/entries/computer-science/> (дата обращения 23.03.2019)